

Amendments To the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1 Claim 1 (canceled).

1 Claim 2 (currently amended): A nonvolatile storage system comprising:
2 a controller capable of receiving commands from a host; and
3 a nonvolatile memory storage coupled to said controller, said nonvolatile memory
4 storage organized into blocks, one or more blocks caused to be identified by a first
5 [predetermined] group of logical block addresses corresponding to a predetermined group of
6 sectors, each sector for including a sector of data and, each block storing two or more sectors
7 of data,
8 wherein said controller, in response to receiving a command from said host to rewrite
9 one or more sectors of data that are stored in said one or more blocks, writes said sectors
10 of data for said one or more sectors of data to be rewritten to one or more new blocks
11 caused to be identified by said first [predetermined] group of logical block addresses
12 [without moving and without copying the sectors of data in the sectors in said one or more
13 blocks that the host did not specify to be rewritten in the command].

1 Claim 3 (currently amended): A nonvolatile storage system comprising:
2 a host for sending commands;
3 a controller coupled to said host for receiving host commands; and
4 nonvolatile memory storage, coupled to said controller, for storing sectors of data
5 into sectors, said nonvolatile memory storage organized into blocks, one or more
6 blocks caused to be identified by a [predetermined] first group of logical block
7 addresses corresponding to a predetermined group of sectors, and each block having
8 two or more sectors of data,

9 wherein said controller receives a command from said host for writing updated one or
10 more sectors of data into a location within the nonvolatile memory storage, said location
11 defined by one or more blocks having previously-written sectors of data, and wherein said
12 controller writes said updated one or more sectors of data into one or more new blocks
13 caused to be identified by said [predetermined] first group of logical block addresses
14 [thereby avoiding moving and without copying all of the previously-written sectors of data
15 every time the host sends a command].

1 Claim 4 (previously presented): A nonvolatile storage system as recited in claim 3 wherein the
2 controller further receives additional commands from the host for further writing, one or more
3 times, sectors of data without moving the previously-written sectors of data every time sectors
4 of data are updated.

1 Claim 5 (currently amended): A nonvolatile storage system as recited in claim 3 wherein the
2 previously-written sectors of data are moved said one or more blocks having previously-
3 written sectors of data at a time later than when the controller writes said updated one or more
4 sectors of data to said one or more new [block] blocks.

1 Claim 6 (previously presented): A nonvolatile storage system as recited in claim 5 wherein
2 said one or more blocks having previously-written sectors of data is erased at a time later than
3 when the previously-written sectors of data are moved from said one or more blocks having
4 previously-written sectors of data.

1 Claim 7 (currently amended): A nonvolatile storage system comprising:
2 a host for sending commands;
3 a controller coupled to said host for receiving host commands; and
4 nonvolatile memory storage coupled to said controller for storing a sector of data,
5 said nonvolatile memory storage organized into blocks, each block having two or more

6 sectors for storing sectors of data, one or more blocks caused to be identified by a first
7 group of [predetermined] logical block addresses corresponding to a predetermined group
8 of sectors, and each block including two or more sectors of data,
9 wherein said controller receives a command from said host for writing updated one
10 or more, but not all, sectors of data into a location within the nonvolatile memory storage
11 said location defined by one or more blocks having previously-written sectors of data, and
12 wherein said controller writes said updated one or more sectors of data to one or more new
13 blocks caused to be identified by said [predetermined] first group of logical block
14 addresses [thereby avoiding moving and copying all the previously-written sectors of data
15 every time the host sends a write command].

1 Claim 8 (previously presented): A nonvolatile storage system as recited in claim 7 wherein
2 the controller further receives additional commands from the host for further writing, one
3 or more times, one or more sectors of data without moving the previously-written sectors
4 of data every time the one or more sectors of data are updated.

1 Claim 9 (previously presented): A nonvolatile storage system as recited in claim 7 wherein
2 the previously-written sectors of data are moved from the one or more blocks having
3 previously-written sectors of data at a time later than when the controller writes said
4 updated one or more sectors of data to said one or more new blocks.

1 Claim 10 (previously presented): A nonvolatile storage system as recited in claim 9
2 wherein the one or more blocks having previously-written sectors of data is erased at a
3 time later than when the previously-written sectors of data are moved from one or more
4 blocks having previously-written sectors of data.

1 Claim 11 (currently amended): A method of updating information in a nonvolatile memory
2 storage having a controller coupled to a host and the nonvolatile memory storage comprising:
3 receiving a command from the host for updating one or more sectors of data into a
4 location within the nonvolatile memory storage, said location defined by a particular block

5 having previously-written one or more sectors of data, said particular block caused to be
6 identified by a [predetermined] first group of logical block addresses, corresponding to a
7 predetermined group of sectors, and including two or more sectors of data, each sector for
8 storing a sector of data;

9 selecting one or more new blocks within the nonvolatile storage; and

10 writing said updated one or more sectors of data to said one or more new blocks
11 caused to be identified by said first [predetermined] group of logical block addresses without
12 moving and without copying the previously-written sectors of data.

1 Claim 12 (previously presented): A method of updating information as recited in claim 11
2 further including the step of receiving additional commands from the host for further
3 updating, one or more times, sectors of data wherein the previously-written sectors of data is
4 not moved every time a sector of data are updated.

1 Claim 13 (previously presened): A method of updating information as recited in claim 11
2 further including the step of moving the previously-written sectors of data from the particular
3 block at a time later than said writing step.

1 Claim 14 (previously presented): A method of updating information as recited in claim 13
2 further including erasing the particular block at a time later than said moving step.

1 Claim 15 (currently amended): A nonvolatile storage system comprising:
2 a controller capable of receiving commands from a host; and
3 a nonvolatile memory storage, coupled to said controller, said nonvolatile memory
4 storage organized into blocks, each block having two or more sectors, each sector for storing a
5 sector of data,
6 wherein said controller, in response to receiving a first write command from the host to
7 rewrite a first sector of data defined by one or more sectors of data that are stored in a
8 particular block, said particular block caused to be identified by a [predetermined] first group
9 of logical block addresses, corresponding to a predetermined group of sectors, and including

10 two or more sectors, writes said first sector of data to one or more new blocks, said one or
11 more new blocks caused to be identified by said [predetermined] first group of logical block
12 addresses, without moving and without copying sectors of data previously-stored in the
13 sectors of said particular block and not specified by the host in the command to be rewritten,
14 said controller, in response to receiving a second write command from the host to rewrite a
15 second sector of data defined by a sector of data within the particular block that is other than
16 the first sector of data, rewrites the second sector of data into the one or more new blocks
17 without moving the first sector of data thereby preventing moving a sector of data every time a
18 write command is received from the host.

1 Claim 16 (currently amended): A nonvolatile storage system comprising:
2 controller coupled to a host for receiving host commands; and
3 nonvolatile memory storage coupled to said controller and organized into blocks
4 having one or more sectors for storing sectors of data, one or more blocks identified by a
5 predetermined group of logical block addresses corresponding to a predetermined group of
6 sectors having stored therein previously-written sectors of data, said controller receiving a
7 host command for writing updated one or more, but not all, sectors of data associated with
8 said predetermined group of sectors into a location within the nonvolatile memory storage
9 defined by said one or more blocks, said controller writing said updated one or more sectors
10 of data to one or more new blocks caused to be identified by said [predetermined] first group
11 of logical block addresses and upon receiving a subsequent command to write subsequent
12 sectors of data associated with said [predetermined] first group of a sectors but that are other
13 than the updated one or more sectors of data, the controller writing said subsequent sectors of
14 data to the one or more new blocks and only moving the previously-written sectors of data if
15 the previously-written sectors of data are not updated.

16
1 Claim 17 (currently amended): A nonvolatile storage system comprising:
2 controller coupled to a host for receiving host commands; and
3 nonvolatile memory storage coupled to said controller and organized into blocks
4 having sectors for storing data, one or more blocks identified by a predetermined group of

5 logical block addresses corresponding to a [predetermined] first group of sectors having stored
6 therein previously-written data, said controller receiving a host command for writing updated
7 one or more, but not all, data associated with said [predetermined] first group of sectors into a
8 location within the nonvolatile memory storage defined by said one or more blocks, said
9 controller writing said updated one or more data to one or more new blocks caused to be
10 identified by said group of logical block addresses and upon receiving a subsequent command
11 to write subsequent data associated with said [predetermined] first group of a sectors but that
12 are other than the updated one or more data, the controller writing said subsequent data to the
13 one or more new blocks and only moving the previously-written data if the previously-written
14 data are not updated.